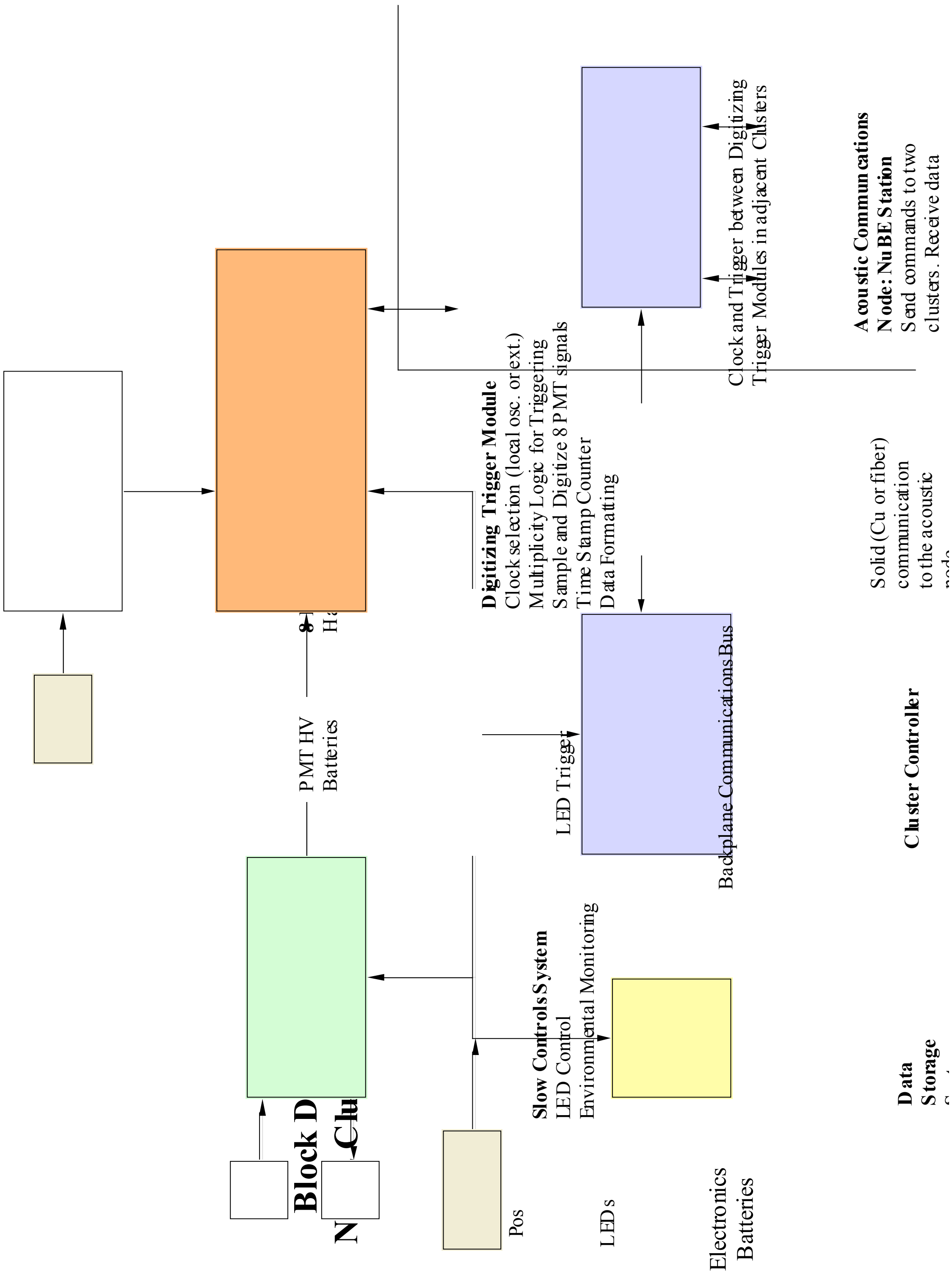
Last change: 14 Nov 2005 HjC

Appendix 1 – NUBE Electronics Details

NUBE Sphere Block Diagram



1. Requirements and Justification for the NUBE Digitizing Trigger Module (DTM)

1.1. Physical Characteristics

1.1.1. Form Factor

Requirement: The DTM must conform to an industry-standard form factor that can be plugged into an industry-standard communications backplane along with the Slow Controls System (SCS), the NUBE Cluster Controller (NCC) and the Data Storage System (DSS). The entire assembly must fit inside a commercial deep-ocean-rated pressure vessel.

Justification: Using industry standard form factors and communications backplanes will simplify the development and production of this module. All the NUBE electronics will be housed in commercially available containers.

1.1.2. Temperature

Requirement: The DTM must be able to operate at temperatures between 14 and 35 degrees Centigrade.

Justification: The ocean temperature at the Nestor/NUBE site is 14 C. During Nestor's March/April tests the temperature inside the sphere leveled out at 29C, and 35C gives us some headroom (see page 25 of http://www.nestor.org.gr/hena_paris/index.html). It should be noted that the NUBE electronics is being optimized to minimize its power consumption (see 1.1.4 below) so the temperature in the NUBE electronics container will level off at a lower value than the 29C obtained by Nestor.

1.1.3. Atmosphere

Requirement: The DTM must operate in a nitrogen atmosphere at 0.25-0.5 atmospheres of pressure.

Justification: The pressure vessel will be filled with nitrogen. It will be less than atmospheric pressure to prevent the two halves of the vessel slipping before they are deployed in the deep ocean.

1.1.4. Power

Requirement: The DTM, and all other pieces of the NUBE electronics, must be designed and implemented in such a way as to minimize their power consumption

Justification: The NUBE electronics will be powered from batteries that have a fixed number of Watt-hours. In order to maximize the deployment time it is necessary to minimize the power usage of the electronics

1.2. Connections

1.2.1. Backplane Communications Bus (BCB)

Requirement: The DTM must connect to the BCB.

Justification: This is the path by which the DTM, SCS and DSS will communicate with the NCC, which is the system responsible for controlling the operation of these modules and reading out their data. The BCB will also provide these modules with their power and ground connections.

1.2.2. PMT signals

Requirement: The DTM must connect to the signal output of 8 PMTs.
Justification: The purpose of this module is to digitize those signals.

1.2.3. Trigger Communications Bus (TCB)

Requirement: The DTM must connect to the DTM of the other cluster in this node via the Trigger Communications Bus (TCB).
Justification: In a NUBE node any hits in one cluster need to be included in the trigger decision of the other cluster. Also, once either DTM has decided to issue a trigger that decision must be passed to the other to initiate digitization and read-out there too.

1.2.4. Slow Controls System (SCS)

Requirement: The DTM must connect to the "LED Trigger" output connector of the SCS.
Justification: The SCS needs to use this connection to force the DTM to sample and digitize the PMT signals, with a known, fixed, start time, when an LED event is initiated.

1.3. Control Functionality

1.3.1. Local Oscillator

Requirements: The DTM must contain a local oscillator that can be used as the global clock source to control the operation of this module. All other clocks used on the DTM must be generated from this global clock source. The oscillator must be accurate to 1 part in 10^7.
Justification: NUBE needs to detect muons from neutrinos in coincidence with other signals detected by satellite based experiments. The clocks in the satellites have accuracy better than 10^{-3} seconds with respect to GMT. We intend to re-sync the local oscillators every 10^4 sec. using the LED system and the clocks on NESTOR.

1.3.2. External Clock

Requirements: The DTM must also have the ability to accept an externally provided clock via the TCB, from the DTM in the other cluster of this node, and use it as the global clock.
Justification: In order for a pair of DTMs to communicate properly for triggering purposes they need to be operating on the same clock frequency. If one DTM sends its local clock to the other then that condition will be met.

1.3.3. Master/Slave Selection

Requirements: The NCC must be able to designate this DTM as either the master of the TCB or a slave.
Justification: This will define which DTM should use its local oscillator as the global clock (the master) and which DTM should use the external clock received from the TCB (the slave).

1.3.4. Master Clock Distribution

Requirement: If this DTM has been designated as the TCB-Master then it must send a copy of its global clock to the other DTM in this node via the TCB

Justification: This will ensure that both the DTMs in both clusters are using the same frequency global clock, which will make it possible for them to communicate properly for triggering purposes.

1.3.5. Slave Clock Monitoring

Requirement: If this DTM has been designated as the TCB-Slave then it must monitor the external clock and switch back to the local oscillator if the external clock fails.

Justification: If the external clock fails then this module will stop working unless it is provided with an alternative clock.

1.3.6. Notification

Requirement: In response to a request from the NCC the DTM must generate a data word that describes the current clock status: “master using local oscillator”, “slave using external clock”, “external clock failed so switched back to local oscillator”, etc….

Justification: The user will need to know which clock is being used.

1.3.7. Synchronous TCB Communication

Requirement: The DTM must use the global clock to control communication with the other DTM over the TCB. Each DTM should drive new data on the TCB synchronized with just one edge (rising or falling, not both) of the global clock. This will guarantee that all information is stable on the TCB for at least one full period of the global clock. Each DTM can then receive new data from the TCB on the other edge (falling or rising) of the clock.

Justification: This will ensure that communication over the TCB is reliable.

1.3.8. Logic Control

Requirement: The DTM must provide the NCC with control of the discriminator threshold on each channel, the Multiplicity Logic Unit (MLU) functionality, the rate at which the scalers are read out and any other aspects of the digitization circuitry that might need to be adjusted.

Justification: The NCC needs to be able to control these aspects of the module’s operation.

1.3.9. Time Stamp Counter

Requirement: The DTM must count cycles of the clock used by the MLU starting from either power-on or an NCC initiated reset. When the MLU issues a trigger the current value of this counter must be included in the data stream. The counter must be large enough that it will not reach its maximum and start over again during a year long deployment.

Justification: This will enable the user to study the time distribution of events. It will also make it possible to cross-calibrate the relative time between NUBE clusters to an integral number of clock ticks. This calibration will be performed using data from LED events, when an LED in a known location is fired with a known duration and amplitude. The requirement that the counter be big enough to not roll over during a year long deployment guarantees that every event will have a unique time stamp, which will make the analysis easier. Please see Section 8.5 of “Implementation of the NUBE DTM” for a description of how many bits the counter will need to meet this requirement.

1.3.10. Priority Flag

Requirement: The DTM must assign a priority flag to this event.
Justification: While NUBE is deployed, a small fraction of the data will be read out acoustically from the DSS. The user will select which subset of events they want to read by specifying a range of priority flags (see Section 2.3.11 of “Requirements and Justification for the NUBE Cluster Controller (NCC)”. The DTM therefore has to assign an appropriate flag to each event before it is read out by the NCC and stored in the DSS.

1.3.11. Event Data Gathering and Formatting

Requirement: Whenever a global trigger is issued the DTM must assemble the data from the 8 PMTs, the time stamp, the priority flag and the full set of trigger information into a formatted event-data package.
Justification: The data needs to be gathered into a known format in order for the user to analyze it.

1.3.12. Scaler Data Gathering and Formatting

Requirement: Periodically, at a rate set by the NCC, the current values of all scalers and any other relevant information (e.g. the time stamp and priority flag) must be gathered into a formatted scaler-data package.
Justification: The data needs to be gathered into a known format in order for the user to analyze it.

1.3.13. Local Memory

Requirement: The DTM needs enough memory to store multiple events internally before they are read-out by the NCC.
Justification: Data will be read out periodically via the NCC. This module needs to store events until they are read out. Please see Section 8.7 of “Implementation of the NUBE DTM” for a discussion of the event size, the event rate, the frequency with which the NCC will read out the data and the resulting size of this memory.

1.3.14. “Busy” Logic

Requirement: The DTM must disable the trigger logic while it is busy processing the last issued trigger, or if there is no more room to store events.
Justification: There is no point in issuing a trigger if it cannot be processed or there is no space to store the data.

1.3.15. Data Transfer to Controller

Requirement: Data must be sent to the NCC on request.
Justification: The NCC is the system responsible for the large-scale storage of the data.

1.4. Digitization Circuitry

1.4.1. Ground and Power Separation

Requirement: The analog and digital power and ground circuits must all be separated from each other. The two ground circuits should meet only at the connection to the BCB ground pin. The two power circuits should meet only at the connection to the BCB power pin.

Justification: This will prevent digital noise being picked up by the analog circuits.

1.4.2. Termination of PMT signals

Requirement: Each of the 8 channels must be correctly terminated to match the impedance of the cable driving it.

Justification: If this is not done correctly the PMT signals will be distorted

1.4.3. Noise

Requirement: The noise generated in the PMT signals by the analog circuitry must be less than 10 mV which is less than 10% of a 120 mV single photoelectron peak (see http://www.nestor.org.gr/hena_paris/index.html on page 36) and is down in the region already dominated by the PMT dark current.

Justification: It must still be possible to distinguish single photoelectron signals from the noise.

1.4.4. Discrimination of PMT Signals

Requirement: Each of the 8 PMT signals must be compared to a threshold which must be useable down to ¼ of a photoelectron, or 30mV. The threshold must be adjustable, on a channel-by-channel basis, by the NCC.

Justification: If we can reliably discriminate when a PMT signal exceeds ¼ of a photoelectron then our ability to detect single photoelectrons approaches 100%. The control is necessary for fine adjustments (if the PMTs are not properly gain-matched then they will produce single photoelectron pulses of different sizes) and in case we need to raise the discrimination threshold on a noisy PMT.

1.4.5. Scalers for Discriminator Output

Requirement: A scaler is needed for each of the 8 channels to count the number of hits on that channel. A hit occurs whenever the PMT signal exceeds its discriminator threshold. Each scaler should be reset to zero after it has been read out.

Justification: It is necessary to monitor the hit rate on each channel individually in order to monitor their performance.

1.4.6. Stretching of Discriminator Outputs

Requirement: The 8 local discriminator outputs must be stretched to be up to 100ns long. The minimum length is the period of the global clock. The actual length must be adjustable by the NCC in units of the period of the global clock. One adjustment for all the signals is sufficient; there is no need to adjust the length for each channel individually.

Justification: 100ns is the maximum time difference between incoming signals in NUBE. It is the sum of the time it takes light to traverse the maximum distance across a node, 12.25m, traveling at ¾ of the speed of light in a vacuum (41 ns) plus the 8 ns worst transit time spread of the PMTs (worst PMT, single p.e.) plus the time it takes the other DTM to send its multiplicity information over 10m of cable to this DTM (~50ns). The minimum stretched pulse length is set by the requirement that the multiplicity information coming from the other DTM must be synchronous with the global clock (see section 1.3.7 above). Since that external multiplicity information is going to have a minimum length of one global clock tick, the local discriminator outputs need to be stretched to the same minimum length to be consistent. Stretching each discriminator output pulse will make it possible to trigger on coincidences between early and late signals from one muon.

1.4.7. Multiplicity Logic

Requirement: The stretched discriminator signals must all go to a configurable MLU that operates using the global clock. Every tick of that clock it will calculate the total multiplicity by counting how many of the local signals are on and adding that to the multiplicity received from the other DTM. The multiplicity is also known as the "coincidence level". The MLU must then produce a trigger signal whenever the total multiplicity exceeds a threshold set by the NCC. It must be possible for the NCC to set up multiple trigger conditions with independent prescales running in parallel. A trigger would be issued whenever any prescale is satisfied.

Justification: This is how the sampling and/or storage of the PMT signals are initiated. The capability of using multiple triggers in parallel is needed in order to make sure rare, important events are triggered while simultaneously continuing to trigger on a subset of very common events.

1.4.8. Local Multiplicity to Other DTM

Requirement: The DTM must send its current local multiplicity to the other DTM. This does not require full resolution; a truncated value (2 bits) indicating 0, 1, 2 or more hits is acceptable.

Justification: The other DTM needs to include any hits from this DTM in its trigger decision. The most likely total multiplicity needed to generate a trigger is 4. Using a 2-bit number to transfer multiplicity information between DTMs will allow the trigger to be correctly generated when each DTM sees 2 local hits. If just 1 bit is used (zero or non-zero multiplicity) then those triggers would be missed. More bits are unnecessary. If any DTM sees a local multiplicity of 4 or more then it will generate a trigger irrespective of any information received from the other DTM. If any DTM sees a local multiplicity of 3 then it will generate a trigger if the multiplicity from the other DTM is non-zero.

1.4.9. Multiplicity Scalers

Requirement: One scaler needs to be implemented for every possible total multiplicity value. Every tick of the clock the scaler corresponding to the current total multiplicity value should be incremented by 1. All of the scalers should be reset to zero after they have been read out.

Justification: It is necessary to monitor the rate at which each coincidence level is occurring.

1.4.10. Global Trigger Decision

Requirement: The MLU trigger must be combined with any trigger received from the other DTM via the TCB and any LED trigger from the SCS to create a global trigger that will initiate readout of the digitized inputs.
Justification: This is how the final trigger decision is made in this module.

1.4.11. <u>Global Trigger to Other DTM</u>

Requirement: If the global trigger is generated from either an MLU trigger or an LED trigger then it must be sent to the other DTM via the TCB to force it to readout its digitized data.
Justification: This is how one DTM will force the other to digitize and readout its data. There is no need to send the global trigger back to the other DTM if the global trigger was created by receiving a trigger from that other DTM in the first place.

1.4.12. <u>Time Resolution of Signal Digitization</u>

Requirement: Once a trigger decision has been made the arrival time of the leading edge of each PMT pulse must be digitized and recorded in such a way that this module contributes no more than 10 ns to the final timing resolution in NUBE.
Justification: 10ns resolution will allow NUBE to reconstruct the velocity vector of high energy, long range, muons that span multiple strings with an angular resolution of 1 degree. This resolution will also make it possible to reconstruct the velocity vector of lower energy, short range, muons that span a single node with an angular resolution of 30 degrees, which is good enough for calibration and monitoring purposes. Please see Section X.Y of the NUBE proposal for a description of how the 10ns resolution was derived.
HANK: Please modify this when you know which section in the proposal justifies the timing resolution.

1.4.13. <u>Depth of Digitization</u>

Requirement: The total time window that will be digitized must be at least 1.53μs.
Justification: The maximum separation between the earliest muon hit and latest is 0.1μs (100ns, see section 1.4.6). The latest time that a useable photon from a slow shower particle can arrive is set by the scattering length of photons in sea water, which is 300m. A photon traveling at c/n = ¾ c will take 1.33μs to travel 300m. Any photon that arrives more than 1.33μs after the trigger probably traveled more than 300m to reach the PMT and is therefore likely to have been scattered at least once. The photon arrival time will no longer provide useful information so that time does not need to be recorded. If the MLU triggers on the earliest muon pulse then it will be necessary to digitize and store the 1.43μs after the trigger. However, if the MLU waits until the latest muon pulse arrives before issuing a trigger then it will be necessary to digitize and store the 0.1μs preceding the trigger and the 1.33μs following it. A time window of at least 1.53μs should cover both possibilities.

1.4.14. <u>Overlapping Pulse Detection</u>

Requirement: It is necessary to be able to detect the existence of multiple overlapping pulses during the time that any one PMT signal is over the discriminator threshold.
Justification: This will improve the ability to reconstruct lower energy muons whose photons arrive at the PMTs very close in time to the photons from the shower particles.

1.4.15. Single-Channel Deadtime

Requirement: Each channel must be dead for no more than 100ns after a pulse crosses the lowest discriminator threshold.
Justification: During the Nestor tests in March/April 2003 the typical single PMT rates were approximately 70 kHz (see page 27 of http://www.nestor.org.gr/hena_paris/index.html). The downward-looking PMTs saw typical rates of 50 kHz (see page 28 of http://www.nestor.org.gr/hena_paris/index.html). We will assume that NUBE PMTs will see similar rates. If each channel is dead for 100ns after its input signal goes over the discriminator threshold then the upward-looking channels will typically be dead for 0.7% of the time, and the downward-looking channels will typically be dead for 0.5% of the time. Let us assume that the hits are mostly random background noise (bioluminescence, potassium-40, etc…). In this case if any channel is dead 0.5% of the time then the probability of the whole cluster being dead is $0.005^8 = 4\times10^{-19}$ (i.e. very small). However, the probability of the whole cluster being live is $(1-0.005)^8 = 96\%$. This should be acceptable.

1.4.16. DTM Deadtime

Requirement: This module must be dead for no more than 10 ms after each trigger issued by the MLU. This includes the time to format the data and store it.
Justification: During the Nestor tests in March/April 2003 the trigger rates were around 3 Hz when the MLU required a 4-fold coincidence and the discriminator thresholds were set to ¼ of a photoelectron (see pages 31 and 39 of http://www.nestor.org.gr/hena_paris/index.html). However, during the commissioning and debugging process triggers that used a lower coincidence level were used, leading to higher trigger rates. We will assume that each NUBE node will encounter similar conditions to the Nestor test. If the module goes dead for 10 ms then the maximum steady state trigger rate than can be achieved is 100Hz which should be big enough.

2. Requirements and Justification for the NUBE Cluster Controller (NCC)

2.1. Physical Characteristics

See "Requirements and Justification for the NUBE Digitizing Trigger Module (DTM)", section 1.1.

2.2. Connections

2.2.1. Backplane Communications Bus (BCB)

Requirement: See "Requirements and Justification for the NUBE Digitizing Trigger Module (DTM)", section 1.2.1.

2.2.2. Node Acoustic Station (NAS)

Requirement: The NCC must have a connection to the NAS of this NUBE node.

Justification: The NAS will be used to communicate with the outside world, both for control and data read-out.

2.3. Functionality

2.3.1. NCC Reprogramming

Requirement: It must be possible for the user to download new programming to the NCC via the NAS.

Justification: It may be necessary to change the operation of the cluster while it is deployed.

2.3.2. DTM Configuration

Requirement: On power-up the NCC must configure any control registers and memories in the DTM using a predefined set of configuration data. It must also be possible for the user to change the configuration set and prompt the NCC to reconfigure the DTM via the acoustic link.

Justification: The NCC is responsible for ensuring that the DTM is operating in the correct way. The user needs the ability to change the configuration since data analysis may indicate that the DTM is not optimally configured.

2.3.3. DTM Read-Out

Requirement: At predefined intervals the NCC must read out all event data stored in the DTM and save it in the DSS. It must be possible for the user to change the read-out interval via the acoustic link.

Justification: The NCC is responsible for ensuring that all the event data is saved locally until it can be read out either via the acoustic link or after the string has been retrieved. The user needs the ability to change the read-out interval since data analysis may indicate that the trigger rate is higher or lower than expected so the read-out rate can be adjusted accordingly.

2.3.4. SCS Power On/Off

Requirement: At predefined intervals the NCC must turn on power to the SCS, and once data has been gathered the SCS must be powered down. For this purpose the SCS should be considered as multiple separate pieces each of which is powered up or down at a rate appropriate to its function. For example, the PMT power and HV control should be on all the time while the acoustic position monitor only needs to be on when it is time to record the position.

Justification: Since NUBE is operating on batteries it is necessary to eliminate non-essential power dissipation to maximize the life of the experiment. It is not necessary for the whole SCS to be switched on all the time.

2.3.5. SCS Configuration

Requirement: Whenever any piece of the SCS is powered up the NCC must configure that piece using a predefined set of configuration data. It must also be possible for the user to change the configuration set and prompt the NCC to reconfigure that piece of the SCS via the acoustic link.

Justification: The NCC is responsible for ensuring that the SCS is operating in the correct way. The user needs the ability to change the configuration since data analysis may indicate that the SCS is not optimally configured.

2.3.6. SCS Read-Out

Requirement: Once any piece of the SCS involved in monitoring has been powered up and configured the NCC must read out all monitor data, form it into a data buffer with the appropriate priority flag (see Section 2.3.11) and save it in local memory.

Justification: The NCC is responsible for ensuring that all the SCS monitor data is saved locally until it can transferred to the DSS for long-term storage.

2.3.7. SCS Data Analysis

Requirement: When monitor data is read out of the SCS the NCC must perform a simple analysis to determine the health of the system being monitored. It must then take the appropriate action if it detects a problem. For example, if the current draw from a PMT rises above some threshold, indicating a leak in a Benthos sphere, then the NCC must shut off all power to that PMT to prevent the batteries being drained. The results of the data analysis, and any action taken, must be saved in the local memory along with the original data.

Justification: Since NUBE will be operating largely autonomously the NCC is responsible for monitoring the basic health of the cluster and responding to a variety of problems. The data must be saved in order for the user to be able to reconstruct the chain of events later.

2.3.8. DSS Power On/Off

Requirement: The NCC must power on the DSS periodically so data from the DTM and the SCS can be transferred to it. Once the data has been stored the NCC must power down the DSS.

Justification: Since NUBE is operating on batteries it is necessary to eliminate non-essential power dissipation to maximize the life of the experiment. It is not necessary for the DSS to be switched on all the time.

2.3.9. DSS Status

Requirement: The NCC must monitor the status of the DSS. If it detects that the DSS is full it must power down all other systems; the DTM, the PMT

power and HV and all other parts of the SCS. These systems should remain off until after the data has been read from the DSS, at which point all other systems can be powered up and data taking can resume.
Justification: Since NUBE is operating on batteries it is necessary to eliminate non-essential power dissipation to maximize the life of the experiment. It is not necessary for the system to be powered up and triggering more events if there is no space to store the data.

2.3.10. Data Transfer

Requirement: On receipt of a command from the acoustic link the NCC must power on the DSS and transmit all the requested data to the NAS. The user will specify the range of priority flags in the command and the NCC must loop through all events currently in the DSS, check the priority flag of each one and transmit only those with flags in the requested range. After the data transfer is complete the DSS can be powered down. The NCC must manage this data transfer in such a way as to minimize the amount of time that the DSS is powered up.
Justification: This is how the data will be read-out while NUBE is deployed.

3. Requirements and Justification for the NUBE Slow Controls System (SCS)

3.1. Physical Characteristics

See "Requirements and Justification for the NUBE Digitizing Trigger Module (DTM)", section 1.1.

3.2. Connections

3.2.1. Backplane Communications Bus (BCB)

Requirement: See "Requirements and Justification for the NUBE Digitizing Trigger Module (DTM)", 1.2.1.

3.2.2. Monitor Inputs

Requirement: The SCS must receive signals from various monitoring devices: acoustic position monitor, humidity and temperature sensors, etc…. The full list is currently undefined.

Justification: These conditions need to be monitored to correctly reconstruct the detector efficiency function.

3.2.3. "Fire" Command to LDBs

Requirement: The SCS must connect to the LDB control input.

Justification: This is used to tell the LDB to actually fire the LEDs.

3.2.4. "LED Trigger" Link to DTMs

Requirement: The SCS must connect to the DTMs "LED Trigger" input.

Justification: The SCS needs to tell the DTM when it has fired the LEDs.

3.3. Functionality

3.3.1. Digitization of Monitor Signals

Requirements: The SCS must provide the ability for the NCC to control the digitization of the analog monitor inputs (e.g. set ADC gate width, initiate digitization, etc…) and read-out the digitized data via the BCB

Justification: Only digitized data can be stored in the DSS, and the monitor data needs to be saved and analyzed along with the PMT data.

3.3.2. LDB Fire Control

Requirement: The SCS must provide the ability for the NCC to set up an automatic sequence of LED firing events specifying the light intensity of each event.

Justification: This is necessary to ensure a regular sequence of calibration events interspersed throughout the data.

3.3.3. "LED Trigger" Command to DTM

Requirement: The SCS must generate a trigger signal for the DTM whenever any LED is fired.

Justification: In order to use the LED system to fully investigate the response characteristics of the system (e.g. effects of light amplitude, cables, differences between PMTs, differences due to different levels of multiplicity logic, etc…) it would be helpful for the LED events to always be triggered on the DTM with a fixed time relationship to the actual LED firing.

4. Requirements and Justification for the NUBE Data Storage System (DSS)

4.1. Physical Characteristics

See "Requirements and Justification for the NUBE Digitizing Trigger Module (DTM)", section 1.1.

4.2. Connections

4.2.1. Backplane Communications Bus (BCB)

Requirement: See "Requirements and Justification for the NUBE Digitizing Trigger Module (DTM)", section 1.2.1.

4.3. Functionality

4.3.1. Stability

Requirement: The DSS must be constructed to hold its data even when the power is switched off.

Justification: The DSS will only be powered up occasionally so the NCC can store new data or read out old data (see sections 2.3.9 and 2.3.11 of "Requirements and Justification for the NUBE Cluster Controller (NCC)". After the access is complete it will be powered down again. The DSS must retain the data when it is powered off so that data can be read out later.

4.3.2. Size

Requirement: The DSS needs to be large enough to hold all of the DTM event and scaler data, and all of the SCS data, from a full year long deployment.

Justification: In case it is not possible to retrieve data via the NAS during the deployment this will ensure that all data is kept until the NUBE string is retrieved. Please see Section 8.7 of "Implementation of the NUBE DTM" for a description of the amount of event and scaler data collected during a 24 hour period. NOTE: I have no estimate yet of the size of the SCS data. However, the DTM will collect 195 Mbytes of data during a 24 hour period. In the 365-day year it will collect ~72 GB of data. To include the SCS data the DSS will hold ~100 GB.

4.3.3. Access Time

Requirement: The read and write access times need to be short enough that the power used by the DSS when it is powered up to be accessed does not exceed its budget (see Section 4.1.2 above).

Justification: The power budget is set by the need to keep the cluster operating during a year long deployment. If the DSS exceeds the power budget then the cluster will exhaust its power supply before the end of the deployment and have to shut down.

5. Requirements and Justification for the NUBE LED Driver Board (LDB)

5.1. Physical Characteristics

5.1.1. Form Factor

Requirement: The LDB must conform to an industry-standard form factor that must fit inside a commercial deep-ocean-rated pressure vessel.
Justification: Using an industry standard form factor will simplify the development and production of this module. All the NUBE electronics will be housed in commercially available containers. Each LDB will be housed in a container separate from the rest of the cluster electronics.

5.1.2. Temperature, Atmosphere and Power

See "Requirements and Justification for the NUBE Digitizing Trigger Module (DTM)", sections 1.1.2, 1.1.3 and 1.1.4.

5.2. Connections

5.2.1. Slow Controls System (SCS)

Requirement: The LDB must connect to the SCS
Justification: The SCS will be used to provide power to the LDB and to control its operation. There will be 2 LDBs per node, one positioned above the node and one below it. The upper LDB will be controlled by the SCS of the upper cluster and the lower LDB by the lower cluster.

5.3. Control Functionality

5.3.1. Stability

Requirement: The control circuitry must be stable against small changes in electrical components.
Justification: This will allow multiple LDBs to be reliably produced. This was a problem for the original Nestor LED driver board.

5.3.2. Amplitude Selection

Requirement: The SCS must be able to specify the amplitude of the LED output.
Justification: This will allow the user to monitor the response of the PMTs to both single and multiple photoelectron pulses.

5.3.3. Pulse Length

Requirement: The control circuitry must be able to produce stable, sharp pulses (1-2ns).
Justification: This is what the pulses generated by muons are like.

5.3.4. Fire LED

Requirement: On receipt of a command from the SCS the LDB must fire the LED with the selected amplitude.
Justification: The purpose of the LDB is to fire the LED to generate a test pulse of light that should be detected by the NUBE PMTs. The data can then

be used to monitor the response of the PMTs to different sized pulses, and to calibrate the relative phase of the clocks in each node.

5.4. LED

5.4.1. Color

Requirement: The LED must emit light with a wavelength as close as possible to 420nm

Justification: The peak of the quantum efficiency of the PMTs is at this wavelength.

5.4.2. Amplitude Range

Requirement: The light output from the LED should be variable over a range where it can produce anywhere from a single photoelectron up to 1000 photoelectrons in the nearest PMT.

Justification: The PMT response to photons from muons varies by a factor of >20, from a 120 mV pulse for a single photoelectron up to 2V, where the output saturates. The LED should be able to stimulate this full range of responses from the PMTs. The LEDs must also be able to generate sufficient light to cause response in the local cluster as wellas in NESTOR, a distance of more than 300m from the LED.

5.5. Angular Distribution

Requirement: The light from the LED system on each cluster needs to illuminate a solid angle of $>3\pi$ steradians.

Justification: This will allow the LED above a node to illuminate all PMTs in all nodes below it and to the side. Similarly the LED below a node will illuminate all PMTS in all nodes above it and to the sides. Detecting photons from an LED fired at a known time will make it possible to synchronize the clocks between clusters in a node and between nodes themselves and NESTOR.

6. Requirements and Justification for the NUBE PMT Bases (PTB)

6.1. Physical Characteristics

6.1.1. Form Factor

Requirement: The PTB must fit inside a 17” Benthos sphere along with a Hamamatsu R2018 PMT

Justification: The PTB is responsible for powering the PMT and providing an interface for its output signal, and all the PMTs are mounted in 17” Benthos spheres.

6.1.2. Temperature, Atmosphere and Power

See “Requirements and Justification for the NUBE Digitizing Trigger Module (DTM)”, sections 1.1.2, 1.1.3 and 1.1.4.

6.2. Connections

6.2.1. NUBE Power Supplies (NPS-HV)

Requirement: The PTB must connect to the NPS-HV

Justification: The NPS-HV will provide the power to operate the PTB.

6.2.2. Photomultiplier Tube (PMT)

Requirement: The PTB must connect to the Hamamatsu PMT connector

Justification: The purpose of the PTB is to provide power to the PMT and an interface to its output signal.

6.2.3. Digitizing Trigger Module (DTM)

Requirement: The PBT must connect to a DTM PMT input connector.

Justification: The DTM is responsible for digitizing and recording the PMT output signal and the PTB must get the signal from the PMT to the DTM.

6.3. Functionality

6.3.1. Input Voltage Regulation

Requirement: The PTB must regulate the voltage it receives from the NPS-HV to stabilize it within 20 Volts of its expected value.

Justification: The final voltage provided to the PMT must be stable or the gain will fluctuate. The 25 volt value assures a gain variation of less than 20%.

6.3.2. Voltage Multiplication

Requirement: The PTB must multiply the voltage it receives from the NPS-HV to make the voltage required by the PMT. That voltage must be adjustable by the user.

Justification: The PMT actually needs a higher voltage than can reasonably be provided by the NPS-HV. The user needs to be able to select which voltage will be provided in order to get the correct gain out of the PMT.

6.3.3. Positive Voltage

Requirement: The PTB must be designed to provide a positive voltage to the PMT, not a negative voltage.

Justification: If the voltage was negative the leakage current through the glass to the ocean (a very large ground) would be enough to rapidly drain the batteries..

6.3.4. Resistive Base Design

Requirement: The PTB must use a resistive base design that limits the current to ~2 μA.

Justification: We expect to use two different variations of the optical module and do not want to develop separate HV multipliers for each dynode chain. It is easier to develop a single HV supply and apply it to a high-resistance dynode chain. The single from a single pe is 10^7e or 1.6×10^{-12} C. Our background rate is $<10^5$ Hz giving a background current of 0.16μA. With a dark current of ~2 μA the background counts will distort the volatge by <1% or <25V for the nominal 2500V setting. This dark current will also lead to a power of <5mw for the PMT.

6.3.5. PMT Output Termination

Requirement: The PTB must provide the PMT with a 50W load into which the PMT can drive its output signal.

Justification: I think this is what the PMT is expecting.

6.3.6. Signal to DTM

Requirement: The PTB must drive the PMT output signal on to a cable that goes to the DTM.

Justification: This is how the signal gets to the DTM.

7. Requirements and Justification for the NUBE Power Supplies (NPS)

7.1. Physical Characteristics

7.1.1. Independence

Requirement: The NPS must consist of batteries.
Justification: NUBE strings will be deployed in the ocean with no connection to land. They must therefore be completely self-contained with an independent source of power for each cluster.

7.1.2. Separation

Requirement: The NPS must be split into 2 separate pieces: the Electronics Batteries (NPS-EB) and the PMT HV Batteries (NPS-HV).
Justification: The electronics modules will require a voltage of the order of a few volts. The PMTs will ultimately need HV of approximately 2500V. The PMT base will accept a lower voltage and multiply that up, however the base design will be greatly simplified if that lower voltage starting point is at least a few hundred volts. Since the electronics modules and the PMTs require such different voltages it is reasonable to provide them with separate supplies.

7.1.3. Housing

Requirement: Both the NPS-EB and the NPS-HV must fit inside commercial deep-ocean-rated pressure vessels.
Justification: All the NUBE electronics will be housed in commercially available containers. Each NPS will be housed in a container separate from the rest of the cluster electronics.

7.1.4. Temperature and Atmosphere

See "Requirements and Justification for the NUBE Digitizing Trigger Module (DTM)", sections 1.1.2 and 1.1.3.

7.2. Connections

7.2.1. Backplane Communications Bus (SCS)

Requirement: The NPS-EB must connect to the BCB
Justification: All of the sphere electronics modules connect to the BCB and draw their power from it..

7.2.2. PMT Bases

Requirement: The NPS-HV must connect to the bases of the 8 PMTs in the cluster.
Justification: Each base will multiply this voltage up to produce the HV needed to operate the PMT.

7.3. Output Voltage

7.3.1. Electronics Batteries

Requirement: The NPS-EB must provide 3.3V .
Justification: This is the standard voltage required by electronics modules.

7.3.2. PMT HV Batteries

Requirement: The NPS-HV must provide a positive voltage equal to at least the minimum dynode stage unit (250V) of the Hamamatsu R2018 PMTs.
Justification: The PMT base will increase this voltage to the required high voltage. Providing the base with the minimum dynode stage unit as a starting point simplifies the base design. The base ultimately needs to produce a positive HV because a negative HV would result in a high enough leakage current through the glass-walled container to the surrounding sea water that the batteries would be drained fast. Since the base needs to generate a positive HV the NPS-HV should provide a positive starting voltage.

7.4. Power

7.4.1. Electronics Batteries

Requirement: The NPS-EB must be able to provide enough continuous power to operate the DTM, the NCC and those parts of the SCS that are powered up continuously for a full year. It must also be able to increase its power output in occasional short bursts to operate the LDB, the DSS and those parts of the SCS that are powered up occasionally. Please see the "Power Consumption" section in each module's implementation description for details on how much power is needed.
Justification: All of these modules need power in order to operate. Each deployment is expected to last for six months, so if the NPS-EB can provide power for a full year there is enough slack in the system in case there is a problem with the string-recovery.

7.4.2. PMT HV Batteries

Requirement: The NPS-HV must be able to provide enough power to operate the PMT HV continuously for a full year.
Justification: The PMTs need power in order to operate. Each deployment is expected to last for six months, so if the NPS-HV can provide power for a full year there is enough slack in the system in case there is a problem with the string-recovery.

8. Requirements and Justification for the Acoustic Communications System (ACS)

8.1. Physical Characteristics

8.1.1. Housing

Requirement: The ACS must be self contained in a housing that can be bolted to the cluster structure.
Justification: The ACS will communicate acoustically through the water to the NESTOR base. There will be one ACS for each cluster.

8.1.2. Power

Requirement: The ACS must have power supplies independent of the NPS.
Justification: There is no reason to share power with this independent system., although all power in NuBE will share a common ground.

8.2. Connections

8.2.1. NUBE Cluster Controllers (NCC)

Requirement: The ACS must connect to the NCC in each of the 16 clusters
Justification: The NCC is the module that is responsible for controlling the operation of each cluster and reading out its data.

8.2.2. Outside World

Requirement: The ACS must connect to the outside world
Justification: The user, in the outside world, is the ultimate destination for the data that is read out, and is also the source of new control commands.

8.3. Functionality

8.3.1. Cluster Selection

Requirement: The ACS must be able to select which of the 2 clusters within which of the 8 NUBE nodes is to be involved in each communication.
Justification: It will be necessary to communicate with each of 16 NUBE clusters individually during deployment.

8.3.2. Bi-directional Communication

Requirement: The ACS must implement a bi-directional communications link between each cluster and the outside world.
Justification: It will be necessary to read out data from each cluster during deployment, and possibly to send new commands if it becomes necessary to change the operation of any cluster.

8.3.3. Data Rates

Requirement: The ACS must relay event and housekeeping data to the NESTOR base at a rate of $>10^{-2}$ Hz. This translates to 10B/s assuming a 1kB event (see sec. 9.7).
Justification: The "real" signal we expect is very small, $<10^{-4}$ Hz, but we will have a much higher "calibration" signal from low energy muons. The ACS will be used to check calibration and detector status as well as to forward the most interesting signals.

9. Implementation of the NUBE DTM

9.1. Introduction

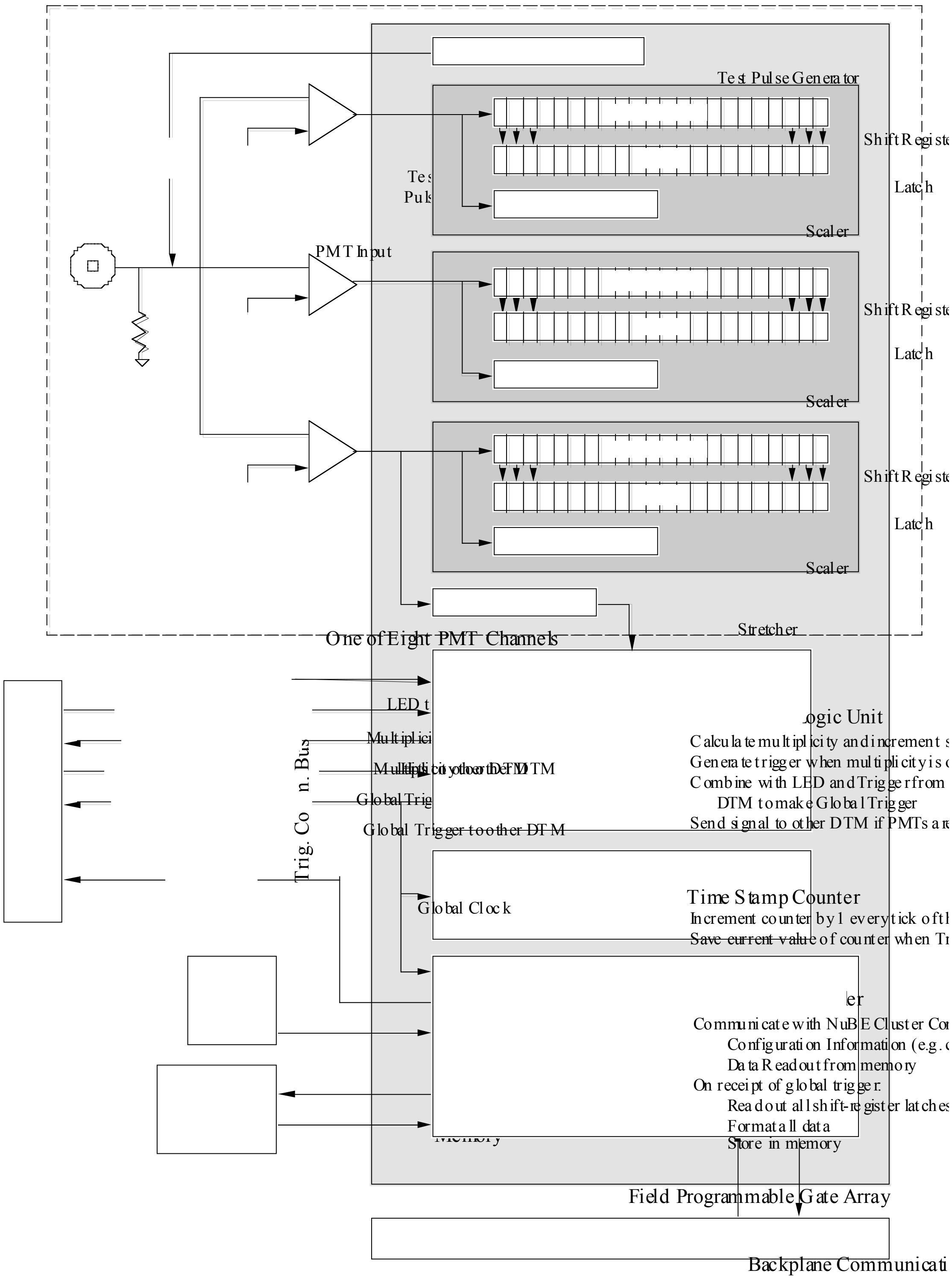


Figure 1: A schematic Diagram of the NUBE DTM

The NUBE DTM is an 8-channel module that uses multiple shift registers, with different thresholds, to record the time at which each input pulse crosses a threshold. Each DTM is designed to operate as one of a pair that communicates for triggering purposes. The modules will be housed in separate electronics spheres in adjacent clusters. A schematic diagram of the module is shown in Figure 1. The module consists of 8 identical input channels designed to be driven by a PMT output, a multiplicity logic unit (MLU), a Time Stamp Counter, a memory block and control logic that provides an interface to the BCB. All the logic is implemented in a Field Programmable Gate Array (FPGA). The various functional blocks will be described in the following sections of this document.

9.2. Clock Distribution

Each DTM will have an ultra-stable 25 MHz quartz oscillator. The DTM can use this local oscillator as the global clock source. Alternatively it can receive an external global clock from the DTM in the other cluster of this node via the TCB. If a DTM is configured to use its local oscillator then it will distribute that clock to the other DTM via the TCB. This 25 MHz global clock will then be used for the Operation Control logic, the MLU and the Time Stamp Counter. The DLLs (Delay-Locked Loops) built into the FPGA will be used to multiply the frequency of the global clock to make a 100 MHz clock for use by all the input channel logic.

9.3. Input Channel

The DTM will implement 8 identical input channels, each of which contains an analog circuit and a digital circuit.

9.3.1. Analog Circuit

Each input will be terminated into 50Ω to provide the correct load for the input pulses. Since the expected pulses in NUBE are already large (1 photoelectron produces a 120mV pulse) there will be no further amplification. Instead the pulse will be immediately driven to the inputs of 3 separate discriminators where it will be compared to three separate threshold voltages. Each threshold voltage will be provided by a user-settable DAC. The outputs of all 3 discriminators will go into the digital circuit in the FPGA. An injection point for a test pulse will be provided just upstream of the split between the 3 discriminators.

9.3.2. Digital Circuit

The digital circuits for all 8 input channels will be implemented inside 1 FPGA. Each circuit will include 3 identical shift registers, a scaler associated with each shift register, a test pulse generator, a stretcher to stretch discriminator pulses from the low-threshold discriminator and a control section

9.3.2.1. Shift Register

The requirement for NUBE is that we be able to achieve a 10ns resolution on the time at which the leading edge of each input pulse crosses the discriminator thresholds. The output of each discriminator will therefore be recorded in a shift register operating at a speed of 100 MHz, so each time bin has a 10ns width. The total depth of digitization is required to be at least 1.53μs so each shift register will contain 160 bins, giving a depth of 1.6μs. Since each bin contains a single bit the size of the shift register is exactly 20 bytes.

When a global trigger is issued the current contents of the shift register will latched into another register, where it will remain until the next global trigger is issued. The data can then be read out from this latched register during the event building process. A shift register does not need to be stopped for its current contents to be latched so it will not contribute to either the single channel dead time or the overall DTM dead time.

9.3.2.2. Scaler

Each discriminator output will also be the input to a scaler that operates at 100 MHz. The scaler will be a series of cascading 16-bit counters that starts from 0 at power-up and can be reset to 0 either by command from the NCC or after it has been read. The current value of the counter will be incremented by 1 whenever there is a hit, i.e. a transition from “0” to “1” is detected in the input. Since the input would have to transition from “1” back to “0” before it could register the next hit, the scaler logic will have a minimum of 2 ticks of the 100 MHz clock to increment the counter.

9.3.2.3. Stretcher

The output of the low-threshold discriminator will also be the input to the stretcher logic. Whenever a transition from “0” to “1” is detected a pulse will be generated with a length defined in a user-settable register. The length can be one of 40ns, 80ns or 120ns, corresponding to 1, 2 or 3 ticks of the 25 MHz global clock. This stretched pulse will be passed to the MLU, which will operate using the 25 MHz global clock.

9.3.2.4. Test Pulse Generator

Logic will be implemented to generate a test pulse for each input that can be fed into the input channel just upstream of the split between the 3 discriminators. Both the duration and amplitude can be specified by the user

9.3.2.5. Control Logic

The final section of each channel is the control logic. This logic will control the latching of the shift register when a trigger is issued. It will save the current value of the scaler, and reset it back to zero, after a pre-determined number of clock ticks. It will control the stretcher length. Finally it will control the test pulse generator.

9.4. Multiplicity Logic Unit (MLU)

The multiplicity logic unit is responsible for generating the global trigger for the DTM. It will be implemented in the FPGA along with the other digital logic. The trigger is generated in two stages. First a multiplicity trigger is generated by combining the stretched discriminator outputs with the multiplicity information received from the other DTM. The multiplicity trigger is then combined with the LED trigger and any trigger received from the other DTM to make the global trigger. Since all information from the other DTM will be received, via the TCB, every tick of the 25 MHz global clock the MLU will operate using that clock too. There will be a scaler for each multiplicity value to count how many times that multiplicity occurs. There will also be shift registers to record the multiplicity information received from the other DTM and the total multiplicity calculated here.

9.4.1. Multiplicity Trigger

The total multiplicity is calculated by summing the number of stretched local discriminator signals that are on (i.e. "1") and adding in the multiplicity information received from the other DTM. A multiplicity trigger will be generated whenever the total multiplicity is over a user-settable threshold. This is shown schematically in Figure 2 for the case where the stretcher length is 120ns.

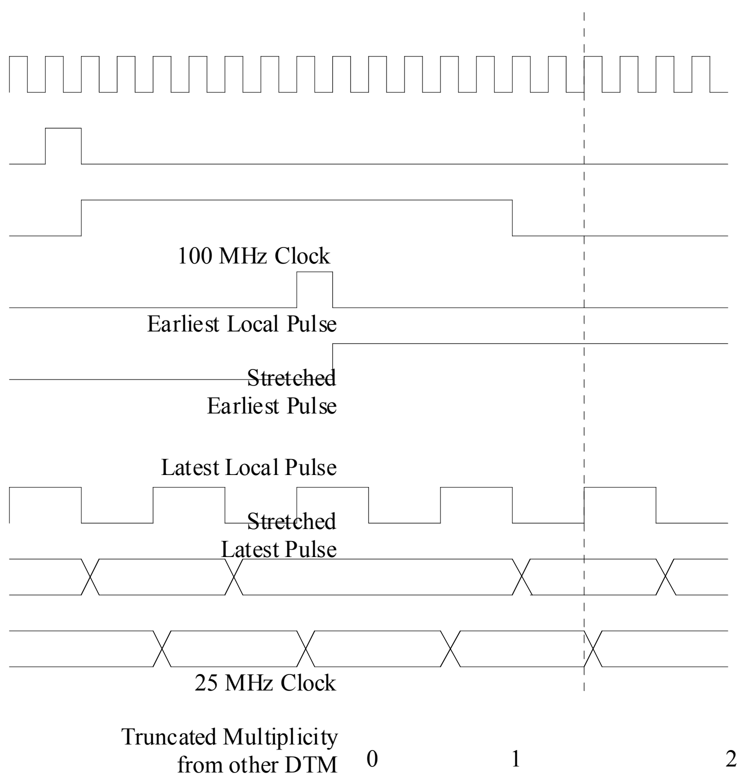


Figure 2: Timing Diagram for DTM Multiplicity Trigger Creation

The multiplicity that led to the trigger will be latched into a register, where it will remain until the next multiplicity trigger is issued. The data can then be read out during the event building process.

9.4.2. Local Multiplicity to Other DTM

A subsection of the total multiplicity calculation is to calculate the local multiplicity by summing the number of stretched local discriminator signals that are on. In parallel with the total multiplicity calculation a copy of the local multiplicity will be truncated to 2 bits meaning the local multiplicity is 0, 1, 2 or more. This truncated local multiplicity will be sent to the other DTM via the TCB.

9.4.3. Multiplicity Scalers

Since each DTM has 8 input channels the local multiplicity can have one of 9 values between 0 and 8. The truncated multiplicity received from the other DTM is a 2-bit number which can therefore have one of 4 values between 0 and 3. Their sum, the total multiplicity, can therefore have one of 12 values between 0 and 11. 12 separate scalers will be implemented, one for each multiplicity value. Every tick of the 25 MHz global clock the scaler corresponding to the current multiplicity value will be incremented by 1. All 12 scalers will be reset to zero after they have been read out, or on receipt of a command from the NCC. Each scaler will be at least a 26-bit counter. 26 bits are sufficient for one counter to increment at 25 MHz for 2.5 seconds (e.g., if the multiplicity is always 0 then scaler 0 will increment every time).

9.4.4. Multiplicity Shift Registers

The total multiplicity, and the truncated multiplicity received from the other DTM, will each be stored in a shift register that operates at the global clock speed of 25 MHz. These shift registers must cover the same 1.6μs total time window as the input channel shift registers. At a speed of 25 MHz, or 40ns per bin, this corresponds to 40 bins. Each bin of the total multiplicity shift register contains a 4-bit number (range 0:11) so the size of that shift register is exactly 20 bytes. Each bin of the truncated multiplicity received from the other DTM contains a 2-bit number so the size of that shift register is exactly 10 bytes.

When a global trigger is issued the current contents of both shift registers will latched into other registers, where the data will remain until the next global trigger is issued. The data can then be read out from these latched registers during the event building process. A shift register does not need to be stopped for its current contents to be latched so they will not contribute to either the single channel dead time or the overall DTM dead time.

9.4.5. Global Trigger

There will be 3 signals that contribute to the global trigger: the multiplicity trigger, the LED trigger from the SCS and any trigger received from the other DTM via the TCB. The three will be combined, i.e. OR'ed together, to create a global trigger. When a global trigger is created the 3 bits will be latched into a register where they will remain until the next global trigger is created. The data can be read out during the event building process.

Once a global trigger has been created the MLU will enter a “busy” state in which generation of further global triggers is blocked until the current event has been fully digitized and readout. The MLU will wait for 1.36μs, i.e. 34 ticks of the 25 MHz global clock, to allow time for hits from slow shower particles to be recorded in the shift registers. After that time the global trigger will be distributed to the input channels, the time stamp counter and the operation control logic.

A subsection of the global trigger logic involves the locally produced signals: the multiplicity trigger and the LED trigger. In parallel with the global trigger logic the OR of these two signals will be sent to the other DTM via the TCB. There is no need to send a trigger received from the other DTM back to it.

9.4.6. Busy Logic

The generation of new triggers will be disabled when the system is busy. This can happen in one of 2 circumstances: the DTM could be busy digitizing and reading out its data from a previous trigger, or the memory could be full so there is nowhere to store any new data. Under normal conditions only the first circumstance should actually occur. In this case the MLU will enter the busy state when it has created a global trigger. It will not leave that state until the operation control logic informs it that all the data has been gathered.

9.5. Time Stamp Counter

In order to combine data from different clusters it is necessary to know as accurately as possible when the trigger was issued in each cluster. This information will be recorded with a time stamp counter. This is a large counter that increments by 1 every tick of the 25 MHz global clock. In this frequency range a 50-bit counter will take over a year to count from 0 to its maximum value. Since typical deployments will last for less than a year this counter will allow each event to be uniquely identified. The current value of the counter will be latched and saved in the data stream whenever a trigger from the MLU is received. This will narrow down the trigger time to within one tick of that global clock. The trigger time will be more tightly constrained within that global clock tick from an analysis of the recorded data.

9.6. Priority Flag

A priority flag will be generated for each event using a look up table (LUT) that has been filled by the NCC. The LUT will have 8 input bits: the 4 bits of the total multiplicity, the 3 bits that are OR’ed together to make the global trigger and 1 bit to indicate if this is a scaler event. Since the LUT has 8 input bits it must have $2^8 = 256$ entries. The output from the LUT will be a 4-bit (1/2 byte) word that can be used by the user to select events that will be read out acoustically. This means the LUT will contain 128 bytes.

9.7. Memory

The DTM will contain enough local memory to store up to 24 hours worth of triggered events and scaler data. Table 1 lists the components of each event type and their size, assuming the scalers are read at a 1 Hz rate. The total comes to around 195 Mbytes which can fit into a commercial 256 Mbytes SIMM (DRAM).

	Triggered Events	Scaler Data	Total
Components	**Header**: 10 bytes **Time Stamp**: 7 bytes (50 bit counter) **Priority Flag**: 1 byte **Input Shift Registers**: 480 bytes (20 bytes/SR, 3 SR/channel, 8 channels/DTM) **Multiplicity Shift Registers**: 30 bytes (total multiplicity and truncated multiplicity from other DTM)	**Header**: 10 bytes **Time Stamp**: 7 bytes (50-bit counter) **Priority Flag**: 1 byte **Input Scalers**: 72 bytes (3 bytes/scaler, 3 scalers/channel, 8 channels/DTM) **Multiplicity Scalers**: 48 bytes (4 bytes/scaler, 12 scalers/MLU, 1 MLU/DTM)	
Event Size	528 bytes	138 bytes	
Event Rate	4 Hz	1 Hz	
Memory used in 24 hours	183 Mbytes	12.0 Mbytes	195 Mbytes

Table 1: Event Sizes in the NDM

The DTM memory will be controlled by the operation control logic in the FPGA. The rate at which new data is written to the DRAM is approximately 2.25 Kbytes/second; 528 bytes/event at a 4 Hz event rate plus 138 bytes/scaler-event at a 1 Hz rate. This is much slower than the DRAM access time, which is approximately 80 Mbytes/second. It will therefore be possible to interleave the read and write operations so they do not interfere with each other. As a result writing new events will not need to be suspended while currently stored events are read out so this memory will have no effect on the dead-time of the DTM.

9.8. Dead-time

The expected dead-time of the DTM is very short compared to the 10ms requirement (see “Requirements and Justification for the NUBE Digitizing Trigger Module (DTM)”, sections 1.4.16). The MLU will introduce 1.36μs of dead-time while it is waiting for hits from slow shower particles to arrive. The shift registers run continuously so they will introduce no dead-time into the system. Of the remaining processes involved in storing data from a triggered event (saving the current time stamp, constructing the priority flag and heaeder and writing all the data into the memory) the memory access is expected to be the limiting time factor. Commercial

DRAM can typically be accessed at 80 Mbytes/second (8 bytes at 10 MHz). At this rate it would take 6.6 μs to write one event of 528 bytes into the memory. The NDM dead-time is therefore expected to be approximately 8 μs, leading to a maximum steady-state event rate of 125 kHz.

9.9. Operation Control

The operation control logic is responsible for configuring the DTM, reading out the data, storing it and communicating with the NCC over the BCB. In response to commands from the NCC it will set the discriminator thresholds, the stretcher length, the properties of the test pulses, and any other user-settable feature of this module. When a global trigger is received from the MLU this logic component will take the appropriate action, e.g. gather the data from the 24 input channel shift registers, the time stamp and all the trigger information, form it into one data block and store it in the local, on-board, memory. Once all the data has been gathered and stored the operation control logic will tell the MLU to leave the "busy" state and continue creating new triggers. At a user-settable rate the operation control logic will also read and clear all the scalers and store that data in the on-board memory. Finally, and again under control of the NCC, it will extract blocks of data from the local memory and pass them to the NCC over the BCB.

9.10. Power Consumption

NOTE: This section needs to be filled in. I need to know the power consumption of:
- the FPGA
- the local memory
- the local oscillator
- each of the 24 discriminators
- each of the 24 DACs
- anything else? Do we need high-power drivers anywhere for the TCB?

Glossary

ACS	Acoustic Communications System
BCB	Backplane Communications Bus
DTM	Digitizing Trigger Module
DSS	Data Storage System
LDB	LED Driver Board
MLU	Multiplicity Logic Unit
NAS	Node Acoustic Station
NCC	NUBE Cluster Controller
NPS	NUBE Power Supplies
NPS-EB	NUBE Power Supplies – Electronics Batteries
NPS-HV	NUBE Power Supplies – PMT HV Batteries
NUBE	Neutrino Burster Experiment
PMT	Photomultiplier Tube
PTB	PMT Base
SCS	Slow Controls System
TCB	Trigger Communications Bus